Soft Sphere Detection with Bounded Search for High-Throughput MIMO Receivers

Predrag Radosavljevic and Joseph R. Cavallaro
Department of Electrical and Computer Engineering
Rice University, Houston, TX 77005, USA
{rpredrag, cavallar}@rice.edu

***Abstract*— [1] We propose a soft sphere detection algorithm where search-bounds are determined based on the distribution of candidates found inside the sphere for different search levels. Detection accuracy of unbounded search is preserved while significant saving of memory space and reduction of latency is achieved. This probabilistic search algorithm provides significantly better frame-error rate performance than the soft K-best solution and has comparable performance and smaller computational complexity than the bounded depth-first search method. Techniques for efficient and flexible architecture design of soft sphere detectors are also presented. The estimated hardware cost is lower than the hardware cost of other soft sphere detectors from the literature, while high detection throughput per channel use is achieved.**

I. Introduction

Future wireless receivers need to support hundreds of MBits/sec data-rates combined with excellent quality of service. In addition, high and flexible spectral efficiency is desired: multiple transmit antennas are accompanied with multiple receiver antennas forming multiple-input multiple-output (MIMO) wireless transceivers [1]. The main challenge, currently, is to design high-throughput low-cost MIMO receivers that can mitigate strong multiple-access interference. Current practical solutions employ detection based on minimum mean square error (MMSE) equalization combined with channel decoding such as low-density parity-check (LDPC) codes [2]. The *a posteriori* probabilities (APPs) of the transmitted coded bits are iteratively improved between outer decoder and inner detector. However, in the case of four transmit/receive antennas and 16-QAM, the error-rate is still far from the capacity limit.

To approach capacity in high spectral efficiency systems, authors in [3] propose approximation of exponentially complex maximum *a posteriori* (MAP) detection: soft sphere detection (SSD). This scheme operates close to channel capacity if iteratively interfaced with outer decoding. However, the detection (search for valid transmitted candidates) is too complex for efficient hardware implementation and design of sub-optimal SSD algorithms have been investigated [4], [5]. We propose the SSD algorithm with bounded search where search bounds are based on distributions of candidates found inside the sphere in different search levels. A substantial reduction of area cost and search latency is achieved while preserving detection accuracy of unbounded search with the same sphere radius. We also propose the cost-efficient and flexible architecture design of soft sphere detectors.

[1]This work was supported in part by Nokia Corporation and by NSF under grants CCF-0541363, CNS-0619767, CNS-0551692, CNS-0321266.

The paper is organized as follows. Section II introduces MIMO system model, soft sphere detection and iterative detection/decoding. The bounded soft sphere detection (BSSD) algorithm is proposed in Section III. Ideas for cost-efficient and flexible architecture design of proposed soft sphere detector are described in Section IV. We conclude the paper in Section V.

II. MIMO Wireless System and Sphere Detection

Coded MIMO wireless system with iterative detection and decoding is shown in Fig. 1. This system is defined by:

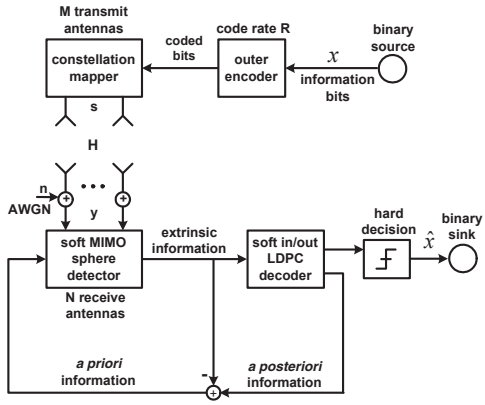

Fig. 1. MIMO transceiver, iterative detection and decoding.

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{1}$$

where M and N are the number of transmit and receive antennas respectively, $\mathbf{y}$ is a vector of N received symbols, $\mathbf{H}$ is the $N \times M$ matrix of flat-fading channel coefficients, $\mathbf{s}$ is a vector of M transmitted modulated symbols, and $\mathbf{n}$ is a vector of additive noise at the receive antennas.

Maximum-likelihood (ML) detection assumes testing of all possible transmit vectors Λ for the minimum square error cost given by $\hat{\mathbf{s}}_{ML} = arg\min_{\mathbf{s}\in\Lambda} ||\mathbf{H}\mathbf{s} - \mathbf{y}||^2$. Since the number of possible candidates is $2^{M \cdot M_C}$ (M_C is the number of bits per constellation symbol), the ML scheme is not suitable for efficient hardware implementation in systems with high spectral efficiency. Sphere detection is a simplification of ML detection where tested candidates are constrained to only those

that are inside a hyper-sphere with a pre-determined radius r around received symbol-vector $\mathbf{y}$ [6]:

$$d(\mathbf{s}) = ||\mathbf{Hs} - \mathbf{y}||^2 \leq r^2. \quad (2)$$

For computationally simpler recursive checking, Eq. 2 is transformed into the identical problem with triangular channel matrix after applying the QR decomposition: $\mathbf{H} = \mathbf{QR}$. The matrix $\mathbf{R}$ is $M \times M$ upper triangular, while matrix $\mathbf{Q}$ is a unitary $N \times M$ matrix. Equation 2 becomes $d(\mathbf{s}) = ||\mathbf{Rs} - \mathbf{Q}^H\mathbf{y}||^2 \leq r^2$. Since matrix $\mathbf{R}$ is upper triangular, the distance $d(\mathbf{s})$ is calculated recursively from one transmit antenna to another:

$$T_m(\mathbf{s}) = T_{m+1}(\mathbf{s}) + \left| \hat{y}_m - \sum_{j=m}^{M} R_{mj} s_j \right|^2 \leq r^2, m = M, \ldots, 1 \quad (3)$$

while $T_{M+1}(\mathbf{s}) = 0$ for all possible transmitted vectors $\mathbf{s} \in \Lambda$. The term $T_i(\mathbf{s})$ is called the partial Euclidian distance (PED) of a candidate symbol $\mathbf{s}$ at the search level m.

The soft sphere detection (SSD) algorithm is an extension of hard sphere detection: instead of only ML candidate it finds the list $\mathcal{L}$ of candidates. The list of candidates provides the reliability information about each transmitted bit x_k which is passed as an extrinsic probability to the outer soft-input soft-output decoder. The extrinsic probabilities can be computed as in [3] using the max-log approximation:

$$\begin{aligned} L_E(x_k|y) \quad &\approx \frac{1}{2} \max_{\mathbf{x} \in \mathcal{L} \cap \mathbb{X}_{k,+1}} \left\{ -\frac{1}{\sigma^2} ||\mathbf{y} - \mathbf{H} \cdot \mathbf{s}||^2 + \mathbf{x}_{[k]}^T \cdot \mathbf{L}_{A,[k]} \right\} \\ &- \frac{1}{2} \max_{\mathbf{x} \in \mathcal{L} \cap \mathbb{X}_{k,-1}} \left\{ -\frac{1}{\sigma^2} ||\mathbf{y} - \mathbf{H} \cdot \mathbf{s}||^2 + \mathbf{x}_{[k]}^T \cdot \mathbf{L}_{A,[k]} \right\}, \quad (4) \end{aligned}$$

where $\mathbf{x}_{[k]}$ is the sub-vector of $\mathbf{x}$ obtained by omitting the k-th bit x_k, $\mathbf{L}_{A,[k]}$ is the vector of all *a priori* probabilities L_A for transmitted vector $\mathbf{x}$ obtained by omitting $L_A(x_k)$, σ^2 is a noise variance, $\mathbb{X}_{k,+1}$ is the set of $2^{M \cdot M_C - 1}$ bits of vector $\mathbf{x}$ with $x_k = +1$, while $\mathbb{X}_{k,-1}$ is similarly defined.

III. Bounded Soft Sphere Detection

Authors in [4] propose hardware design of the soft sphere detection with the depth-first search with up to 256 search operations. One search operation is defined as a simultaneous testing of all constellation points if they are inside the hyper-sphere based on the cumulative distance from the previous search levels (see Eq. 3). This solution achieves good performance but the latency is long because of the large number of search operations. The soft K-best scheme from [5] is suitable for pipelining because of the constant latency per search level and therefore achieves high detection throughput. On the other hand it suffers substantial performance degradation because a large number of good candidates are discarded at every search level.

In order to preserve accuracy of the search process, we utilize a variable maximum number of candidates per search level, unlike the K-best approach where the search-bound per level is fixed to the K best candidates. The goal is also to avoid sorting of candidates after every search level and to lower the probability of discarding good candidates in early search levels. Figure 2 shows the principle of proposed bounded soft sphere detection (BSSD). The maximum number of valid candidates per search level is pre-determined. When the search-bound is reached or exceeded the search process stops and continues in the next search level. Candidates that exceed predetermined bounds are discarded from the search process. The number of search operations is upper-bounded by $1+\sum_{l=1}^{M-1} Max_l$ which pre-determines the largest detection latency.

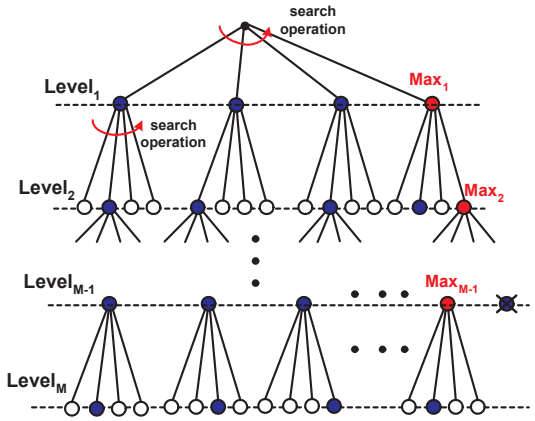

Fig. 2. The tree-search visualization of bounded search with variable bounds per search level: bounded soft sphere detection BSSD(Max_1,Max_2,...,Max_{M-1}). M transmit antennas, QPSK.

Search-bounds are determined in such a way to preserve detection accuracy of unbounded search with the same radius value. Furthermore, the goal is to reduce memory space for the storage of partial/final candidates/distances as well as to substantially decrease latency of the search process. Figure 3 shows distributions of the number of candidates inside the hyper-sphere for different search levels if the number of candidates is bounded in both the second and third search levels to 100. Frequency non-selective channels from Rayleigh distribution are assumed, as well as a 4x4 16-QAM system with unitary transmit energy. The sphere radius of 0.6 is fixed and pre-determined to provide on average about 25 final candidates if the search is not bounded. At the beginning, the majority of tested candidates are inside the sphere: PEDs are small and candidates are unreliable. This is why the K-best solution from [5] suffers performance degradation: probability of discarding good candidates is high even after the sorting of PEDs. Therefore, we don't limit the number of valid candidates in the first search level. However, the PEDs monotonically increase during the search process and percentage-wise more candidates in the lower search levels are outside the sphere. While the number of candidates in the first search level cannot be bounded without significant performance degradation, the search process in lower search levels can be stopped once a certain number of candidates is reached and still preserve detection accuracy. In this case, a significantly smaller number of partial and final candidates

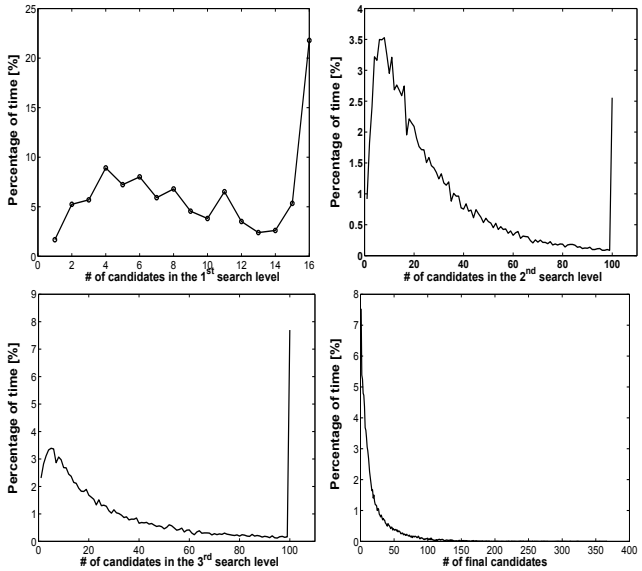

Fig. 3. Distribution of valid candidates in the 2^{nd}, 3^{rd} and 4^{th} search level: bounded search, radius=0.6, Rayleigh fading channels, 4x4 16-QAM. Maximum number of candidates is 100 in both 2^{nd} and 3^{rd} search levels.

is observed than for the unbounded search. Therefore, the storage of partial/final candidates and distances requires a much smaller memory.

Figure 4 shows the frame error rate (FER) performance for different detection schemes. Identical system parameters are assumed as in the distribution analysis. The outer LDPC decoding with 15 inner iterations is used, the codeword size is 1944, code rate is 1/2, and there are four outer iterations between detection and decoding. The proposed detection approach is labeled as BSSD(16,100,100) according to the bounds for different search levels. It can be observed that only a small loss occurs compared to unbounded soft sphere detection (USSD), with significant gain compared to MMSE with decision feedback equalization (MMSE-DFE) from [7] and soft K-best from [5]. Our approach has comparable performance with the depth-first search from [4] with reduced complexity due to better utilization of valid candidates in the 1^{st} search level: 64 search operations on average vs. 256. Furthermore, latency of the BSSD approach is reduced compare to the USSD where approximately 82 search operations are performed on average for the chosen sphere radius of 0.6.

IV. Architecture Design of BSSD

In this section we present the high-level architecture design of the physical layer MIMO receiver based on iterative improvement of the *a posteriori* probabilities between the inner bounded soft sphere detection (BSSD) and outer LDPC decoding. A high level block diagram of the MIMO receiver is shown in Fig. 5. Search function unit searches for candidates that are inside the pre-defined hyper-sphere based on Eq. 3. Valid candidates and corresponding PEDs are stored in the search memory and used in the lower search levels. Once the

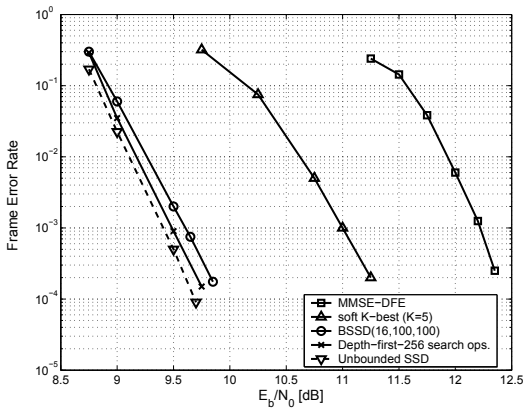

Fig. 4. Frame error rate performance of proposed bounded soft sphere detection BSSD(16,100,100) vs. other soft detection schemes.

final candidates/distances are determined they are passed to the *a posteriori* probability (APP) function unit that computes the reliability information of coded bits according to Eq. 4. The List memory stores all lists of final candidates within the current LDPC codeword: number of stored lists is $C_S/(M \cdot M_C)$, where C_S is the LDPC codeword size. Search for the list of final candidates is performed only in the initial iteration. Then outer feedback is employed between LDPC decoder and APP function unit to improve *a posteriori* information based on updated *a priories*. In the following subsections, we analyze in more detail the architecture implementation of subcomponents of the soft sphere detector.

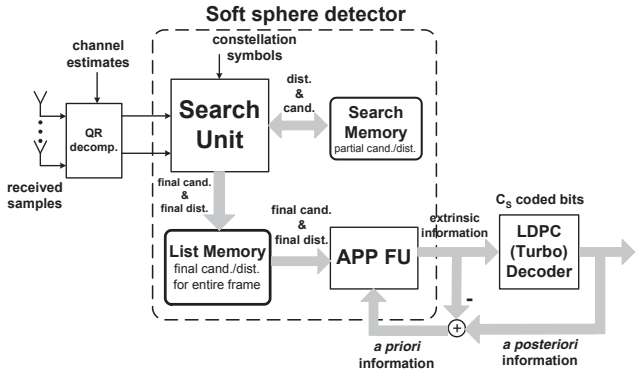

Fig. 5. Physical layer MIMO receiver with iterative detection and decoding.

A. Arithmetic Logic

The arithmetic logic is composed of two components: the search function unit and the APP function unit. It is assumed M=4 transmit antennas in the system. The search function unit simultaneously computes PEDs for $P_C = 2^{M_C}$ constellation points that correspond to the particular symbol-candidate found in the previous search level. The search unit then checks if PEDs are within the sphere radius and saves all valid candidates (up to P_C) as well as their PEDs in the search memory for later use. Final candidates and the corresponding

final Euclidian distances are directly utilized by the APP function unit and stored in the List memory for use in outer feedback.

For the sake of clarity we repeat Eq. 3 assuming (without losing generality) that the current search level is the final and computationally most complex one: $T_1(\mathbf{s}) = T_2(\mathbf{s}) + \left|\hat{y}_1 - R_{11}s_1 - \sum_{j=2}^{M=4} R_{1n}c_j\right|^2 \leq r^2$, while c_2, c_3, and c_4 are candidates saved in the search memory for the 2^{nd}, 3^{rd} and 4^{th} transmit antenna respectively, $T_2(\mathbf{s})$ is the corresponding PED, and s_1 represents any of P_C constellation points that are simultaneously tested. Factors F_m=$\hat{y}_m - R_{mm} \cdot s_m$ (for $m = M, ..., 1$) are pre-computed for all M search levels, stored in registers and used for testing of all candidates. In addition, it is not necessary to compute all $2 \cdot P_C$ products $R_{mm} \cdot s_m$ per search level: only four for 16-QAM and eight for 64-QAM since there are four and eight different signal levels, respectively. Furthermore, rather than computing products, it is simpler to apply shift/add operations on R_{mm} values because of the known levels of constellation points.

For parallel computation of products $R_{mj} \cdot c_j$ $(j = 2, 3, 4)$ twelve function units FU_{CSA} and six add/subtract arithmetic units are required. The FU_{CSA} checks the value of partial candidate c_j loaded from the memory and then performs the appropriate shifting, add/subtract operation and sign-conversion. The block diagram of the arithmetic logic for computation of a single factor $R_{mj} \cdot c_j$ is shown in Fig. 6. Computation of Euclidian distances can now be rewritten

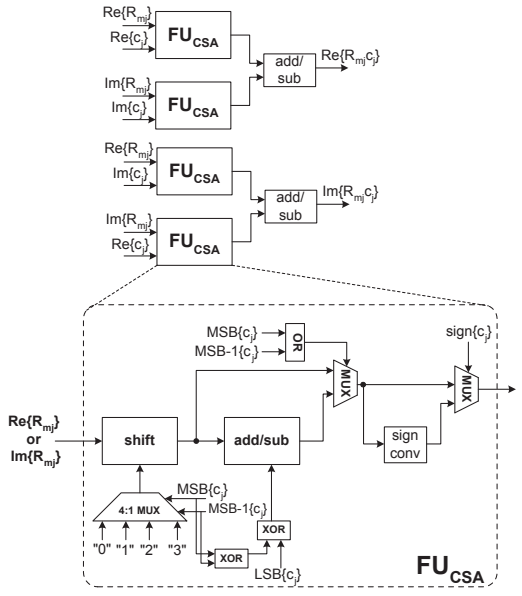

Fig. 6. Block diagram of arithmetic logic for computation of $R_{mj}c_j$ for a single value of j. Block diagram of the FU_{CSA} for computation of partial $R_{mj}c_j$ products: support for both 16-QAM and 64-QAM.

as: $T_1(\mathbf{s}) = T_2(\mathbf{s}) + |Re\{X\} + iIm\{X\}|^2 \leq r^2$, where $X = F_1 - \sum_{j=2}^{4} R_{1j}c_j$. There are only four and eight different values of $Re\{X\}$ and $Im\{X\}$ for 16-QAM and 64-QAM, respectively. Therefore, parallel computation of $|X|^2$ requires 8 and 16 square multipliers for the two cases. P_C different values of $|X|^2$ will be formed by the cross-addition of all individual results of square multipliers as shown in Fig. 7.

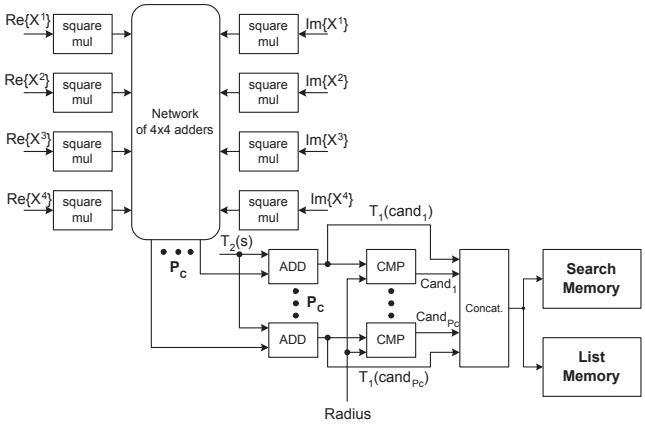

Fig. 7. Final stage of the search function unit: parallel square operations, computation of the P_C PEDs by cross-addition and comparison with the sphere radius.

The APP function unit updates the *a posteriori* probabilities of $M \cdot M_C$ coded bits according to Eq. 4. It can be noticed from Eq. 4 that the computed Euclidian distance $T_1(\mathbf{s})$ of the vector-candidate $\mathbf{s}$ can be reused. The total inner product using sign-conversion and additions is first computed and then the appropriate $L_A(x_k)$ is excluded: $\mathbf{x}_{[k]}^T \cdot \mathbf{L}_{A,[k]} = \mathbf{x}^T \cdot \mathbf{L}_A - sign(x_k) \cdot L_A(x_k), \forall k = 1, ..., M \cdot M_C$. Support for both 16-QAM and 64-QAM can be insured by dedicating enough parallel arithmetic units. Updated $M \cdot M_C$ extrinsic probabilities $L_E(x_k|\mathbf{y})$ will be directly passed to the memory of the LDPC decoder without any buffering.

B. Memory Organization

The memory space is divided into two parts: the search memory and the List memory. For cost-efficient design one-ported search memory is proposed. All candidates found inside the sphere per search operation and their PEDs are concatenated and stored in a single memory location. The depth of the search memory is equal to the number of search operations, while the width initially corresponds to the constellation size. In order to reduce the memory width we divide the search memory into modules: one module for each search level except the last one. In all search levels except the first one, a much smaller number of candidates than P_C are found inside the sphere per search operation. This means that the effective width of the search modules for the 2^{nd} and 3^{rd} levels can be significantly reduced. By applying this approach, a substantial reduction of the search memory is achieved: from 35 KBits to 16 KBits for 16-QAM and from 335 KBits to 48 KBits for 64-QAM.

Each address location of the List memory contains the final candidate ($M \cdot M_C$ bits) and the Euclidian distance with a precision of b bits. The depth of the List memory corresponds to the number of channel realizations per data-frame and to the average number of candidates for all channel realizations. The

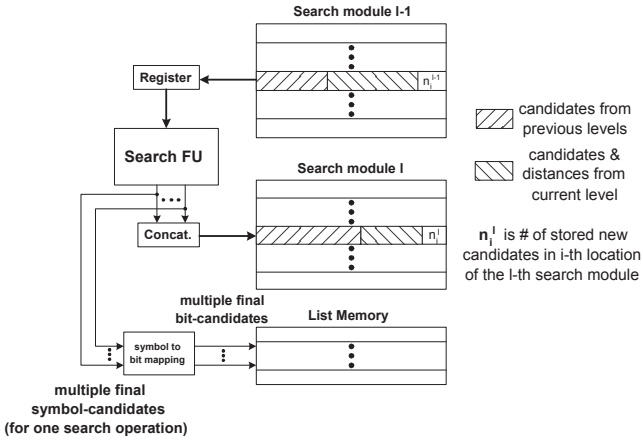

Fig. 8. Interface between search memory modules, search function unit and the List memory.

memory depth needs to support the worst case which is defined as the maximum of all average number of final candidates among the large number of considered channel realizations. The size of the List memory is $\left\lceil \frac{C_S}{M \cdot M_C} \cdot Max\{Av.Cand_f\} \right\rceil \times (b + M \cdot M_C)$, where $Max\{Av.Cand_f\}$ is the maximum among all average number of candidates per data-frame f.

Figure 8 shows the interface between two search memory modules, the search function unit and the List memory. The content of a particular search module is stored in the register and appropriate sub-parts are loaded to the search FU - one candidate from level l-1 at a time combined with partial vector-candidates from a total of l-2 previous search levels. If the last search-level is reached, final candidates and distances are directly stored in the multi-port List memory after de-mapping of symbol-candidates into bit-candidates.

C. Estimated Hardware Cost

Table I shows the estimated number of standard CMOS logic gates for design of the search FU, the APP FU and the exact memory size. Both arithmetic units support 16-QAM and 64-QAM modulations. We assume 12-bit fixed-point precision for arithmetic operations, and representation of final Euclidian distances. If parallel soft sphere detectors are employed for simultaneous detection of multiple channel realizations, the arithmetic logic and the search memory space needs to be parallelized while a single List memory is used since it is dedicated for the entire data-frame.

TABLE I

ESTIMATED HARDWARE COST FOR DETECTION OF ONE CHANNEL USE.

	4x4 16-QAM	4x4 64-QAM
PED FU	27 K gates	27 K gates
APP FU	17 K gates	17 K gates
Search Memory	16 KBits	48 KBits
List Memory (frame)	103 KBits	144 KBits

Table II shows the estimated detection throughput, area cost and frame-error rate performance of the proposed BSSD(16,100,100) solution vs. two hardware implementations from the literature: depth-first SSD with 256 search operations from [4], and soft K-best detector from [5]. Soft detection of four 16-QAM symbols at a time is assumed. The area estimates for our solution correspond to the Chartered 0.13 μm CMOS technology [8] library where the size of a two-input NAND gate is 4.5 μm^2 and the cell area of one-port memory is 3.61 μm^2. An implementation overhead of 30% is also included. The estimated latency of one search operation is about 2 clock cycles. Our design solution has the smallest area cost because of the memory reduction and efficient design of the arithmetic logic. Furthermore, our solution is significantly faster than the solution from [4] with similar detection accuracy.

TABLE II

COMPARISON OF SOFT SPHERE DETECTORS.

F_{CLK}=200 MHz	[4]	[5]	BSSD(16,100,100)
Throughput [MBits/sec]	7.68	106	≈ 25
Area [mm^2]	0.88	0.56	≈ 0.34
SNR [dB] @ FER=10^{-3}	9.5	11	9.6

V. CONCLUSION

We propose soft sphere detection with bounded search based on the distribution of candidates found inside the sphere in different search levels. This scheme has reduced memory requirements and smaller latency compare to unbounded search while preserving detection accuracy. The frame-error rate performance is much better compare to the soft K-best from [5] and the computational complexity is smaller than the bounded depth-first search from [4]. Furthermore, we propose ideas for area-efficient and flexible architecture design: the estimated hardware cost is smaller than two previous architecture solutions while achieving high detection throughput.

REFERENCES

[1] G.J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, vol. 1, pp. 41–59, 1996.

[2] B. Lu, G. Yue, and X. Wang, "Performance analysis and design optimization of LDPC-coded MIMO OFDM systems," *IEEE Transactions on Signal Processing [see also IEEE Transactions on Acoustics, Speech, and Signal Processing]*, vol. 52, pp. 348–361, Feb. 2004.

[3] B.M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Transactions on Communications*, vol. 51, pp. 389–399, March 2003.

[4] D. Garrett, L. Davis, S. ten Brink, B. Hochwald, and G. Knagge, "Silicon complexity for maximum likelihood MIMO detection using spherical decoding," *IEEE Journal of Solid-State Circuits*, vol. 39, pp. 1544–1552, Sept. 2004.

[5] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 491–503, March 2006.

[6] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, pp. 463–471, Apr. 1985.

[7] B. Lu, G. Yue, and X. Wang, "Performance analysis and design optimization of LDPC coded MIMO OFDM systems," *IEEE Transactions on Signal Processing*, vol. 52, pp. 348–361, Feb. 2004.

[8] "Chartered semiconductor manufacturing technology; 0.13-micron solutions." http://www.charteredsemi.com/technology/13_micron.asp.